Automated Indications and Warnings from Live Surveillance Data

Paul J. Metzger

A software system called the Fusion Exploitation Tools (FET) underlies an automated process for alerting intelligence analysts when abnormal activity occurs in an area under their surveillance. The FET consists of algorithms that analyze vehicle and ship traffic. It works in conjunction with the Google Earth geospatial visualization system, presenting the user with a distributed network-centric way of visualizing this information. The FET is unique in that it automatically analyzes live intelligence data streams in near real time. The FET also incorporates direct machine-to-machine requests for additional data collection, providing both enhanced situation awareness and a timely and effective means to control sensor platforms.

Intelligence exploitation is the process by which raw intelligence data are analyzed and the meaning behind the data is determined. Historically, intelligence data have been imagery (either electro-optical, infrared, or radar), and exploitation was manually performed by analysts. These analysts would interpret and report on important aspects of the imagery, such as a submarine leaving a Soviet naval base or a collection of tanks readying for a field exercise. In recent years, imagery has been augmented with a different form of intelligence data, called moving-target indication (MTI). MTI provides the position and velocity of vehicles and ships that are moving. One of the challenges with MTI data is the extremely large volume of information; thus, the manual exploitation of MTI is highly labor intensive and time consuming.

The Fusion Exploitation Tools (FET), developed at Lincoln Laboratory under the sponsorship of the National Geospatial-Intelligence Agency (NGA), is a network-centric system that performs automated exploitation in near real time. The goal of the FET is to ingest large amounts of intelligence, surveillance, and reconnaissance (ISR) data and automatically cue analysts (or sensors) when anomalous activity is detected. Consider the example of the screenshot in Figure 1. It's a Google Earth view generated by the FET, and it shows several regions under surveillance where automated MTI analysis indicates normal activity (green) and several regions where abnormal activity was detected (red). In this example, the presence of several convoys and fast-moving vehicles that

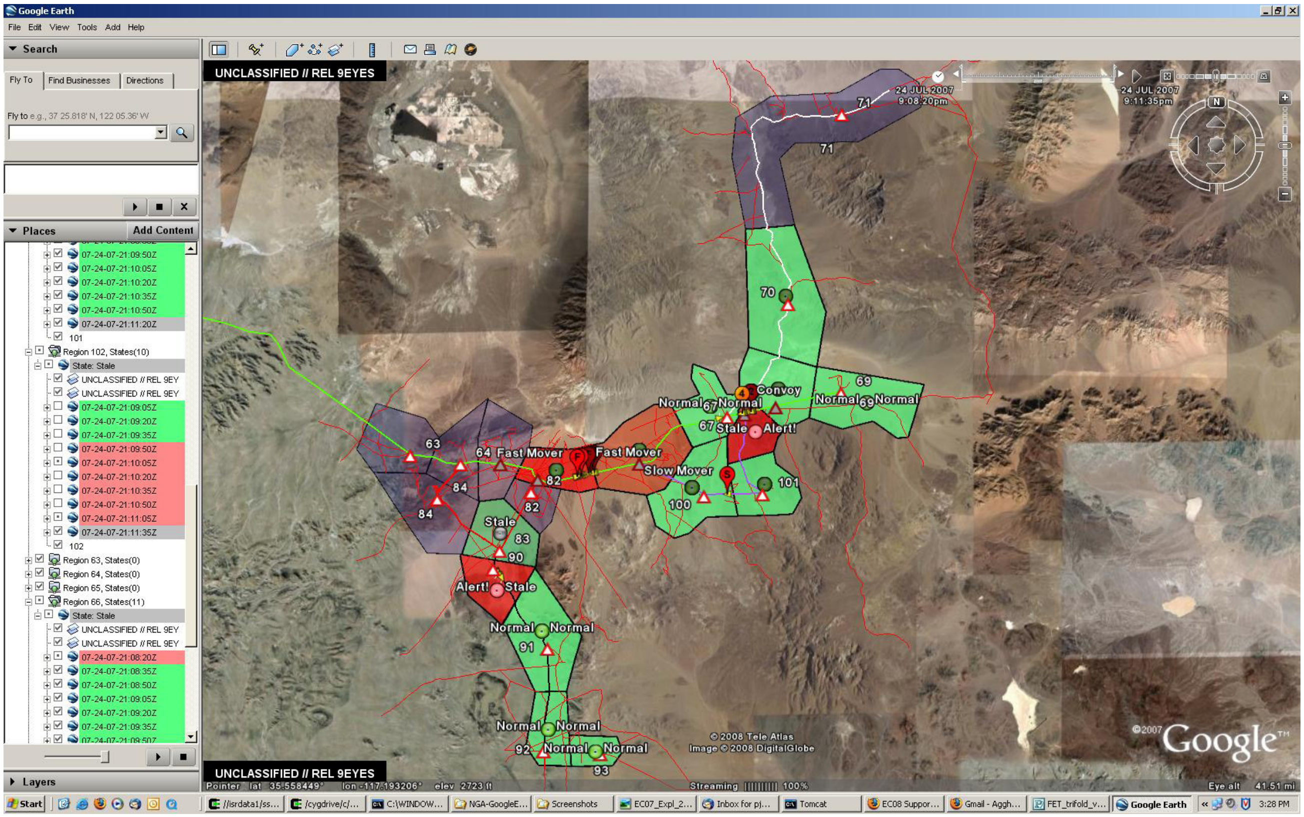


FIGURE 1. A Google Earth view shows situational awareness via the Fusion Exploitation Tools (FET) watchboxes. Blue areas are regions under surveillance for which no intelligence, surveillance, and reconnaissance (ISR) data are yet available. Green areas indicate that normal activity was detected, while red areas indicate warnings that activity is outside the scope of previously observed normal behavior for that area. The panel at left provides a color-coded historical view of the state of each watchbox over time.

were outside the bounds of normal traffic patterns was an anomaly that triggered alerts.

The ability of the United States and its coalition forces to develop and deploy sensors has dramatically increased the need for systems that can help operators quickly extract knowledge and understanding from the ever-increasing volumes of sensor data produced. In January 2007, NGA and the Department of Defense's Space Radar integrated program office approached Lincoln Laboratory and asked for concepts on how a network-centric automated exploitation system could assist with these challenges. At that time, the NGA had begun exploring Google Earth as a common framework within which to display ISR data and had recently deployed Google Earth servers on both the Secret Internet Protocol Router Network (SIPRNet) and the Joint Worldwide Intelligence Communications System. Thus the development of the FET began, with an initial emphasis on automated exploitation of MTI data, an architecture designed for a growth path supporting ingest of additional ISR data types, and a design goal for a display system based on Google Earth. FET was to turn Google Earth into a dynamically updating situation-awareness display (Figure 1), automatically generating indications and warnings in near real time as live ISR data streams were received.

An example of how the FET can be used is to monitor suspicious areas for the presence of a convoy. Such monitoring is typically employed when an attack from a vehicle-borne improvised explosive device has begun. By automatically detecting a convoy, the FET could cue the operator that this group of vehicles may be suspicious. If the operator believes that this situation warrants further investigation, he or she can use the FET Convoy Alert to automatically request an unmanned aerial vehicle (UAV) to take a closer look.

Several MTI analysis tools already exist. The most popular of these are the Moving Target Exploitation system (MTIX, and PC MTIX), DotMatrix, and the ISR

About Google Earth

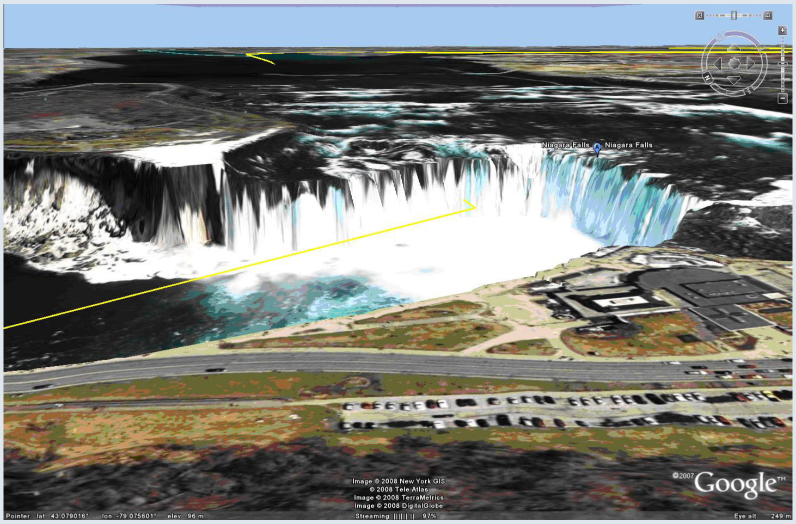

Niagara Falls
Niagara Falls
Image © 2008 New York GIS
© 2008 Tele Atlas
Image © 2008 TerraMetrics
Image © 2008 DigitalGlobe
©2007 Google™
Pointer lat 43.079016° lon -79.075601° elev 96 m
Streaming ||||||:|| 97%
Eye alt 249 m

FIGURE A. This Google Earth image of Niagara Falls is an example of the type of three-dimensional imagery available as a backdrop to the situational-awareness layers that can be provided by the FET.

Google Earth is an interactive, three-dimensional data viewer originally developed by Keyhole, Inc. Centralized servers stream the virtual globe, comprising underlying geographical data, to the Google Earth client software.

Google Earth supports the display of spatiotemporal data layered on top of the underlying globe data, known as the Google Earth "data brick." The version of the data brick that Google provides on the Internet has numerous interesting built-in layers above and beyond the basic capabilities of satellite imagery draped over digital terrain elevation data (earth surface height information). It includes such basic geographic information as roads, city names, and state and country boundaries.

One powerful feature of Google Earth is that its maps can link to content created by individual users, such as Wikipedia entries, digital photos, and geospatially tagged YouTube video. Toward this end, Google Earth provides for an entire custom layer of user-specified data, defined via the Keyhole Markup Language (KML), an XML schema specific to Google Earth. These KML files allow a user to specify geometric primitives (points, lines, polygons), imagery overlays, and temporal data (the ability to play back time-sequence data). They also provide the ability to have the Google Earth client software periodically poll a server to check for the arrival of new KML content. This function is achieved via the KML NetworkLink concept.

Forensics Tool. Figure 2 and Figure 3 show displays of data from the Lincoln Multimission ISR Testbed (LiMIT) by PC MTIX and DotMatrix, respectively. All of these are stand-alone software tools that allow analysts to investigate MTI data sets. None of these tools, however, were designed to perform automated near-real-time exploitation or analysis. In fact, the FET has demonstrated utility as a cueing system for these other tools. In other words, the FET can be used as a first-order filter to cue an analyst where abnormal activity is happening. Once cued, the operator then uses one of the above MTI analysis tools to investigate further.

How FET Works

The FET is a web application server (or portal) that works in conjunction with the capabilities provided by the Google Earth client. The FET does not need to be installed on individual users' computers. Instead, all of the features of the FET are available via the network through a standard web browser. The ability to add or delete ISR data from the system is restricted to users with administrative rights, but all users can view the live or archived data, or customize the view according to their needs. The software is written primarily in Java and operates within the framework of the Apache Tomcat system.

The system consists of three large areas of functionality: data ingest, automated exploitation, and display update. Data ingest is performed in a variety of ways, all of which use standard ISR data formats.

Data Ingest. In the first data-ingest approach, the FET monitors a directory (or shared network folder) for the arrival of new data. This method is used when a sensor generates file-based MTI reports. Consider an example in which data are being produced by LiMIT. As data are collected, new MTI files are generated and sent wirelessly to the LiMIT ground station. There the data files are stored in a folder that is constantly monitored by the FET. The saved files are automatically read and parsed, and their contents are stored in a relational database. (The current implementation of the FET uses the searchable geospatial database PostgreSQL.) The second approach is to listen to the network for ISR data broadcasts. As in the file-based approach, data received through the

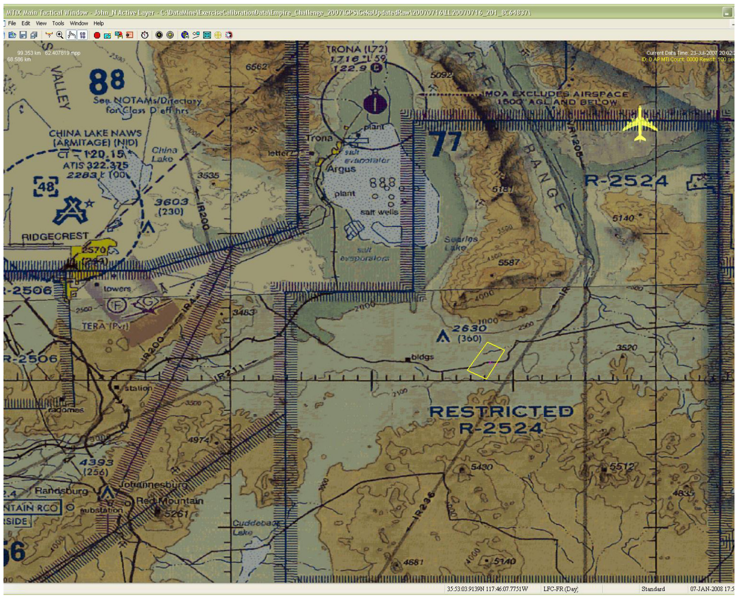
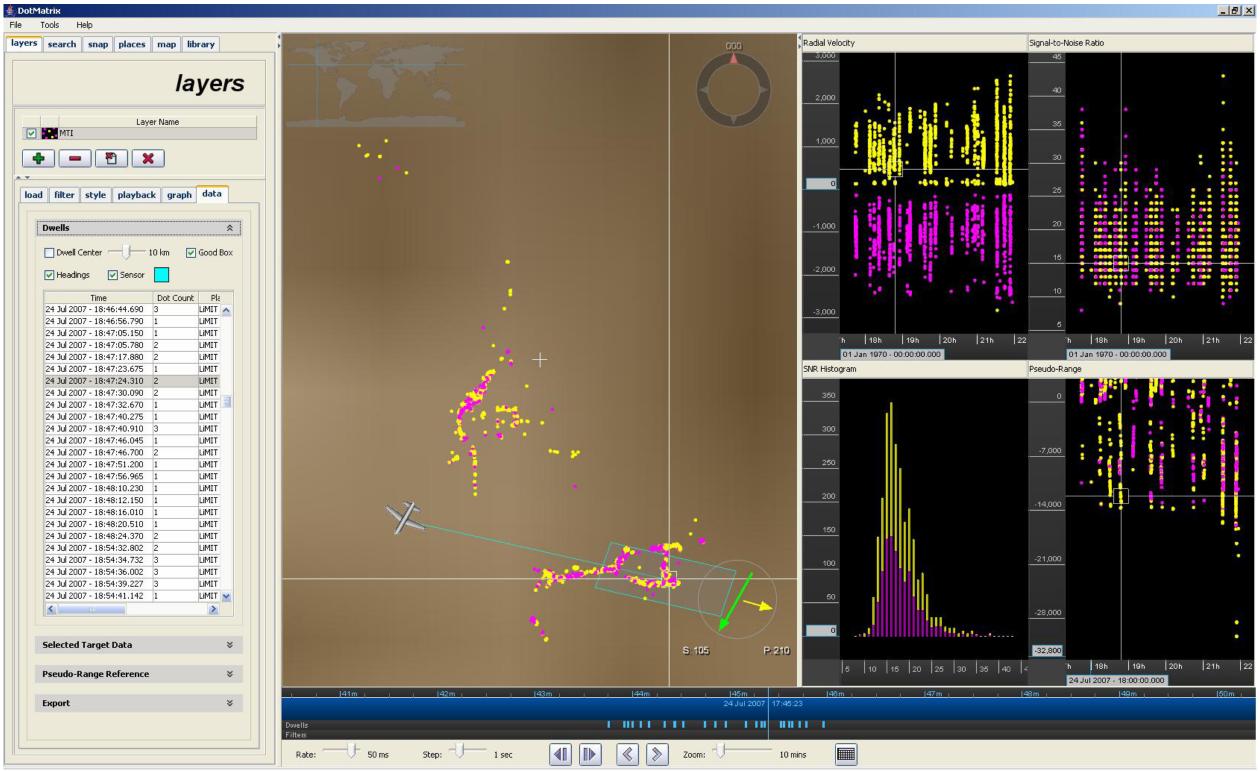


FIGURE 2. The PC MTIX Display Client (MDC) is a common tool used for performing moving-target indication (MTI) analysis. The screenshot shows the position of the Lincoln Multimission ISR Testbed (LiMIT) aircraft (yellow aircraft icon, upper right) at the point in time it was collecting a single MTI radar beam dwell (yellow rectangle, center) during the Empire Challenge 2007 field exercise.



☑ MTI

Dwells

□ Dwell Center 10 km ☑ Good Box

☑ Headings ☑ Sensor

Time	Dot Count	Pla
24 Jul 2007 - 18:46:44.690	3	LiMIT
24 Jul 2007 - 18:46:56.790	1	LiMIT
24 Jul 2007 - 18:47:05.150	1	LiMIT
24 Jul 2007 - 18:47:05.780	2	LiMIT
24 Jul 2007 - 18:47:17.880	2	LiMIT
24 Jul 2007 - 18:47:23.675	1	LiMIT
24 Jul 2007 - 18:47:24.310	2	LiMIT
24 Jul 2007 - 18:47:30.090	2	LiMIT
24 Jul 2007 - 18:47:32.670	1	LiMIT
24 Jul 2007 - 18:47:40.275	1	LiMIT
24 Jul 2007 - 18:47:40.910	3	LiMIT
24 Jul 2007 - 18:47:46.045	1	LiMIT
24 Jul 2007 - 18:47:46.700	2	LiMIT
24 Jul 2007 - 18:47:51.200	1	LiMIT
24 Jul 2007 - 18:47:56.965	1	LiMIT
24 Jul 2007 - 18:48:10.230	1	LiMIT
24 Jul 2007 - 18:48:12.150	1	LiMIT
24 Jul 2007 - 18:48:16.010	1	LiMIT
24 Jul 2007 - 18:48:20.510	1	LiMIT
24 Jul 2007 - 18:48:24.370	2	LiMIT
24 Jul 2007 - 18:54:32.802	2	LiMIT
24 Jul 2007 - 18:54:34.732	3	LiMIT
24 Jul 2007 - 18:54:36.002	3	LiMIT
24 Jul 2007 - 18:54:39.227	3	LiMIT
24 Jul 2007 - 18:54:41.142	1	LiMIT

Rate: 50 ms Step: 1 sec Zoom: 10 mins

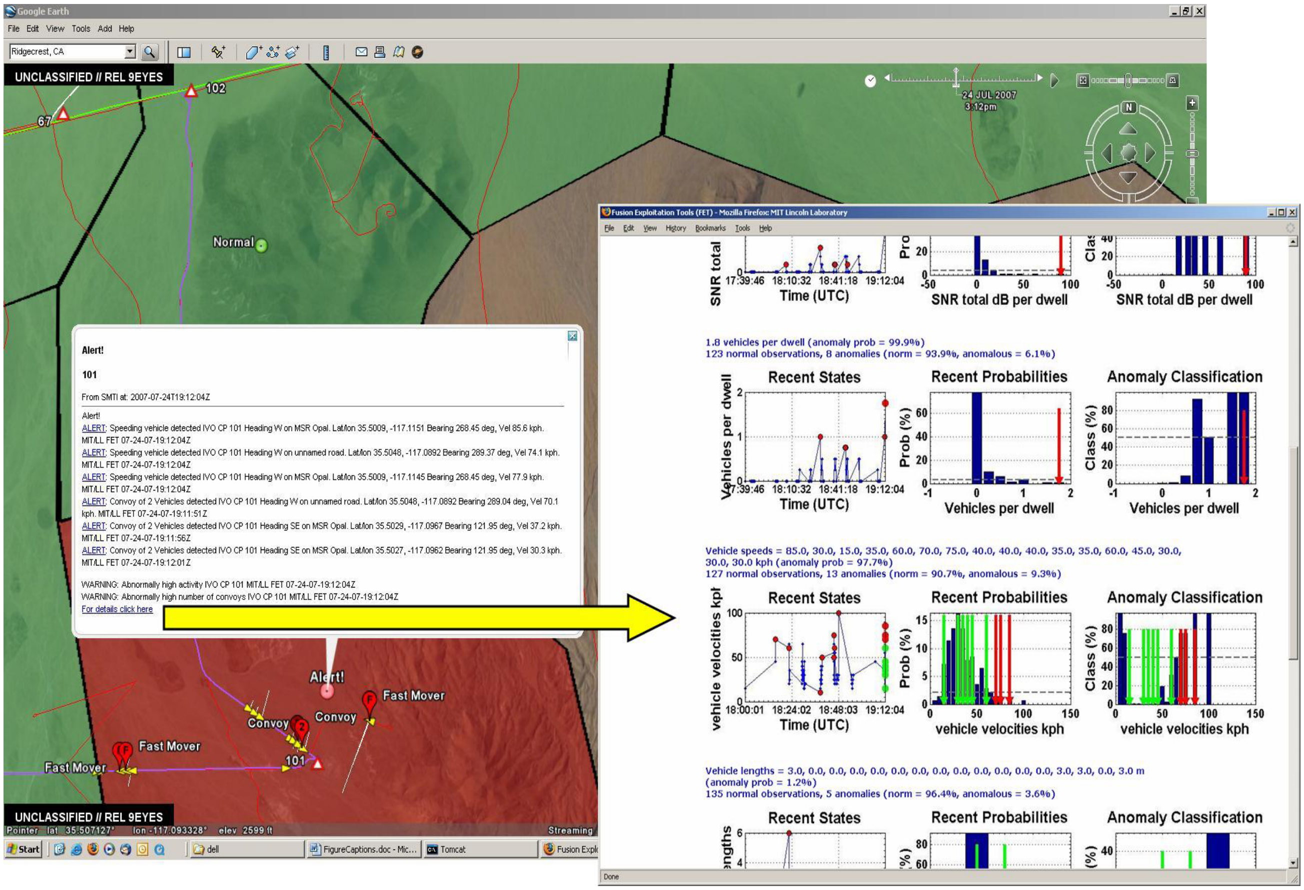
FIGURE 3. The DotMatrix tool (shown viewing LiMIT radar MTI data) can be used to perform detailed analysis of individual MTI detections. However, with the large volume of MTI data regularly collected, a tool like the FET is needed to help the analyst focus on what subset of MTI to investigate.

network are processed and stored in the database. Both of these approaches are fully automated—once the FET has been started and the proper parsers have been loaded, no user interaction is required. The third approach for delivering data to the FET is to use a standard web-based upload form that allows the user to explicitly load the desired data into the system.

Having three different ways of importing data gives the FET significant flexibility, and it can operate well in just about any network architecture. Moreover, FET's ability to ingest data in all three manners makes it compatible with a wide range of sensor systems. For example, the Joint Surveillance Target Attack Radar System (JSTARS) is an aircraft that generates MTI and streams it over the network; it does not generate file-based MTI. LiMIT, on the other hand, is file based and does not stream data over the network. By supporting both types of data feed, FET can work with both JSTARS and LiMIT (as well as numerous other MTI sources).

Exploitation Process. The exploitation process runs asynchronously to the data ingest, allowing the analysis algorithms to execute while new data are being read and stored in the database. The exploitation algorithms use a sliding time window to analyze chunks of MTI data—typically 15 seconds' worth at a time. The algorithms are implemented in Matlab and operate by monitoring areas of interest, called watchboxes. These watchboxes are defined in Google Earth, dropped into a special folder (or upload form) on the FET, and stored in the database. When the Matlab exploitation process is started, it loads the watchboxes. As it periodically polls the database, the system is on the lookout for the appearance of new ISR data in each watchbox.

In Figure 1, the watchboxes are updated as data arrive. In this example, the Region 82 watchbox—the red area near the center of the screenshot—has detected abnormal activity (fast movers), while the green regions indicate normal activity and the blue areas show that no



Alert!

101

From SMTI at: 2007-07-24T19:12:04Z

Alert!
ALERT: Speeding vehicle detected IVO CP 101 Heading W on MSR Opal. Lat/lon 35.5009, -117.1151 Bearing 268.45 deg, Vel 85.6 kph. MIT/LL FET 07-24-07-19:12:04Z
ALERT: Speeding vehicle detected IVO CP 101 Heading W on unnamed road. Lat/lon 35.5048, -117.0892 Bearing 289.37 deg, Vel 74.1 kph. MIT/LL FET 07-24-07-19:12:04Z
ALERT: Speeding vehicle detected IVO CP 101 Heading W on MSR Opal. Lat/lon 35.5009, -117.1145 Bearing 268.45 deg, Vel 77.9 kph. MIT/LL FET 07-24-07-19:12:04Z
ALERT: Convoy of 2 Vehicles detected IVO CP 101 Heading W on unnamed road. Lat/lon 35.5048, -117.0892 Bearing 289.04 deg, Vel 70.1 kph. MIT/LL FET 07-24-07-19:11:51Z
ALERT: Convoy of 2 Vehicles detected IVO CP 101 Heading SE on MSR Opal. Lat/lon 35.5029, -117.0967 Bearing 121.95 deg, Vel 37.2 kph. MIT/LL FET 07-24-07-19:11:56Z
ALERT: Convoy of 2 Vehicles detected IVO CP 101 Heading SE on MSR Opal. Lat/lon 35.5027, -117.0962 Bearing 121.95 deg, Vel 30.3 kph. MIT/LL FET 07-24-07-19:12:01Z

WARNING: Abnormally high activity IVO CP 101 MIT/LL FET 07-24-07-19:12:04Z
WARNING: Abnormally high number of convoys IVO CP 101 MIT/LL FET 07-24-07-19:12:04Z
For details click here

1.8 vehicles per dwell (anomaly prob = 99.9%)
123 normal observations, 8 anomalies (norm = 93.9%, anomalous = 6.1%)

Vehicle speeds = 85.0, 30.0, 15.0, 35.0, 60.0, 70.0, 75.0, 40.0, 40.0, 40.0, 35.0, 35.0, 60.0, 45.0, 30.0, 30.0, 30.0 kph (anomaly prob = 97.7%)
127 normal observations, 13 anomalies (norm = 90.7%, anomalous = 9.3%)

Vehicle lengths = 3.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 3.0, 3.0, 0.0, 3.0 m (anomaly prob = 1.2%)
135 normal observations, 5 anomalies (norm = 96.4%, anomalous = 3.6%)

FIGURE 4. When a target is highlighted, textual information is presented. A pop-up window shows statistical details of FET exploitation algorithm calculations on demand.

Cursor on Target

Cursor on target (CoT) has emerged as a standard XML messaging format for sharing situation-awareness information. The FET currently does not exploit CoT data. However, there does exist a CoT-to-KML converter that allows Google Earth to display CoT state data (e.g., position, orientation, and identification) in conjunction with FET data. CoT also defines a message format for requesting additional ISR data. In addition to supporting the automated machine-to-machine CoT task request and having the ability to generate CoT tracks, the FET also displays the current status of a CoT task (whether it's been acknowledged, whether the sensor has agreed to collect the data, and whether the data collection has been completed).

current information is available. As the algorithms process new data, they compare the current information with statistical data previously calculated for that watchbox and determine whether the new data are within normal statistical limits.

Because average vehicle speed is one component of the analysis, different regions can have different statistical limits. A vehicle traveling 45 mph may be considered normal in one region, but in another it may trigger an alert that the vehicle is traveling faster than average. One way that FET stands alone among MTI analysis tools is that the user has access to all of the statistical information that underlies the reasoning behind an alert. When an alert is generated, the details are displayed in a Google Earth pop-up message (Figure 4). The pop-up also contains a hyperlink that leads to a web page containing statistical charts (generated on demand) detailing the rationale behind the alert. The results of the exploitation algorithms are then time stamped and stored back into the database, and the results are converted into Keyhole Markup Language (KML) for display in Google Earth.

Display Update. Key to FET's value is its ability to dynamically update Google Earth. The core of the design revolves around the KML NetworkLink construct. Two broad categories of information utilize this approach in the FET—the display of raw ISR data as they arrive, and the dynamic update of exploitation results.

For raw data, especially for large-volume data such as that produced by MTI, the FET needs a way to limit the volume of data loaded into Google Earth and to update the data. The approach that we chose uses the concept of a regular grid overlaid on the exercise area, as shown in Figure 5. This grid is itself a KML file and is loaded from the FET web portal (new grids can be easily generated from the portal to support new exercise areas). Each cell in the grid contains a NetworkLink that periodically polls the FET for new data. At a user-defined interval (the fastest refresh available is 1 s), Google Earth attempts to reload a KML file specified by the NetworkLink. Upon receiving this request, the FET regenerates the KML content for that grid cell and passes the information back to Google Earth, which then displays the new results.

In the grid-cell construct, the visual display of the grid works independently of the data update, allowing the user to disable the visual grid but to retain the functionality of dynamic data updates. This structure allows the screen to be less cluttered while maintaining the dynamic update capability. The other useful feature of the data grid is the ability to indicate that data are available without requiring the user to see all of the underlying atomic data elements: cells for which data are available are shaded yellow. This capability is another mechanism to reduce screen clutter, and it guarantees that the user is presented raw data only in grid cells that are of interest.

The dynamic update of exploitation results works in a similar fashion. The watchboxes are originally created

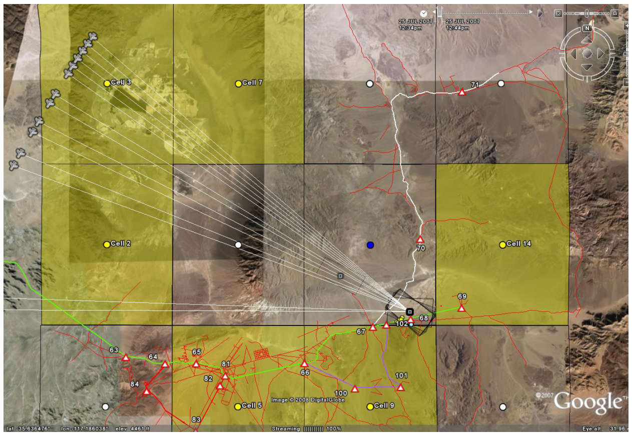
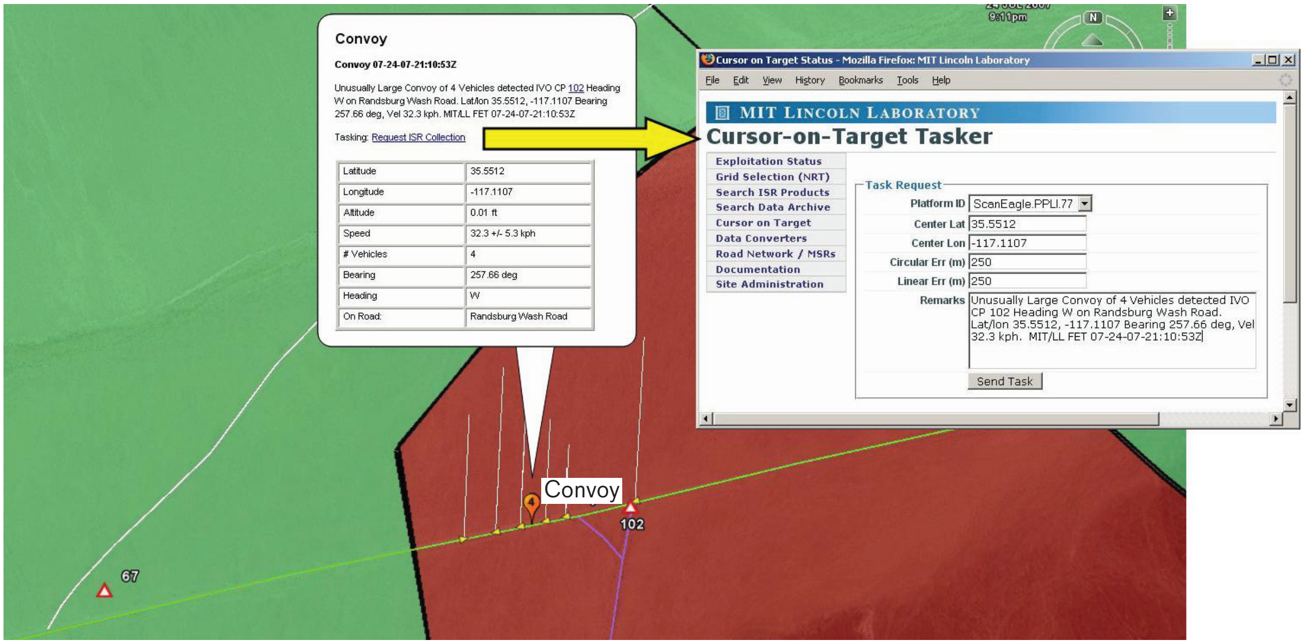


FIGURE 5. Six yellow grid cells show presence of MTI; one cell (2nd row, 3rd column) is enabled to show raw MTI data.



Convoy

Convoy 07-24-07-21:10:53Z

Unusually Large Convoy of 4 Vehicles detected IVO CP 102 Heading W on Randsburg Wash Road. Lat/lon 35.5512, -117.1107 Bearing 257.66 deg, Vel 32.3 kph. MIT/LL FET 07-24-07-21:10:53Z

Tasking: Request ISR Collection

Latitude	35.5512
Longitude	-117.1107
Altitude	0.01 ft
Speed	32.3 +/- 5.3 kph
# Vehicles	4
Bearing	257.66 deg
Heading	W
On Road:	Randsburg Wash Road

Cursor on Target Status - Mozilla Firefox: MIT Lincoln Laboratory

MIT LINCOLN LABORATORY

Cursor-on-Target Tasker

Task Request

Platform ID: ScanEagle.PPLI.77

Center Lat: 35.5512

Center Lon: -117.1107

Circular Err (m): 250

Linear Err (m): 250

Remarks: Unusually Large Convoy of 4 Vehicles detected IVO CP 102 Heading W on Randsburg Wash Road. Lat/lon 35.5512, -117.1107 Bearing 257.66 deg, Vel 32.3 kph. MIT/LL FET 07-24-07-21:10:53Z

FIGURE 6. The FET makes the process of performing a machine-to-machine tip and cue extremely easy. Operators click on the alert (the four-vehicle convoy detection shown above) and manually review the automated warning message from the FET. If they determine that they would like additional sensor coverage, they click on the Request ISR Collection link, which brings up a task request form (with the fields automatically populated by the FET). The final step is to select the sensor platform from a drop-down list of available resources and click Send Task.

in Google Earth, then loaded into the FET, where they have the associated NetworkLink control mechanisms added. The user then loads these watchboxes via the FET web portal, and the NetworkLink objects poll the FET in exactly the same way that the data grid cells do. The visual result is watchboxes that change color as new exploitation results are calculated.

When the FET generates an indication or warning, it generates a piece of KML code that uses the Placemark object—a clickable Google Earth icon that the user can interrogate for detailed information on the exploitation. The Placemark provides two paths for dissemination of the exploitation results.

In the first path, the exploitation result is textual data, along the lines of a military situation report that contains the essential elements of information: what's happening, where it's happening, and who's reporting the information. The user can then copy and paste the text data into e-mail or online chat windows.

The second way to disseminate the exploitation information via a Placemark connects the automated exploitation process with a semiautomated collection request. In this path, the FET exploitation result contains a hyperlink to a cursor-on-target (CoT) task request form using the standard CoT message format (see the sidebar, "Cursor on Target" on page 71). Clicking this hyperlink populates all of the essential information on the tasking request, as shown in Figure 6. This approach allows the user to easily perform a machine-to-machine tasking request, providing a mechanism to cue narrow-field-of-view sensors (such as a UAV) from wide-field-of-view data such as MTI.

A recent addition to the FET provides yet a third mechanism to disseminate results. This manual process involves the generation of a track—a series of spatiotemporal reports associated with a specific vehicle—that is then sent onto the network via CoT. When FET users see indications that a suspicious vehicle is moving in an area of interest, they can pull up a web form and fill out the fields (latitude, longitude, track ID, and comments) and send the CoT track out onto the network (FET automatically time-stamps the data). Contact information that is sent out via a CoT track will appear in any application that ingests CoT data, including FalconView—a geographical-data-display software package used widely by the Department of Defense. This method allows FET results to be sent to users who don't have access to the Google Earth viewer.

Moving-Target Indication

MTI reports are an ISR data product provided by numerous radar sensors (including Lincoln Laboratory's own LiMIT radar, which flies aboard a modified Boeing 707). MTI data are normally disseminated in a standard format (known as STANAG 4607) and viewed through one of the MTI analysis tools previously described. In addition to being able to display raw MTI, the FET runs several additional algorithms on the data.

The first algorithm takes the position of the raw MTI detection (the coordinate of the MTI detection is generally referred to as the MTI dot) and projects it onto the nearest road. MTI generally has very small measurement errors along the line of sight to the sensor. However, cross-range errors—that is, perpendicular to the sensor—can be on the order of a football field or more. Such imprecision makes it difficult to interpret a display of MTI dots. The FET road projection starts at the position of the MTI dot and begins projecting in both directions along the cross-range error segment until the first road segment is encountered. This will be the position of the projected dot.

Knowing the line-of-sight velocity (which is in the original MTI data report) and the orientation of the road permits calculation of the velocity component along the road, giving both a direction of travel on the road and an estimated true road speed. The FET display indicates direction of travel by using an oriented triangle. In existing MTI analysis tools, the color of the MTI dot indicates movement toward the sensor (magenta) or away from it (yellow). However, in a multisensor environment, and even in an environment where the sensor is orbiting, this color scheme becomes confusing. In the FET, therefore, the color of the MTI dot (both the raw dot and the projected dot triangle) is based on the signal-to-noise ratio (SNR) of the detection. High SNR values (12 dB and above) indicate strong detections and are likely to be vehicles, which get color-coded yellow. Low SNRs (4 dB and below) are possible false alarms, color-coded red. Values in between are shown in orange.

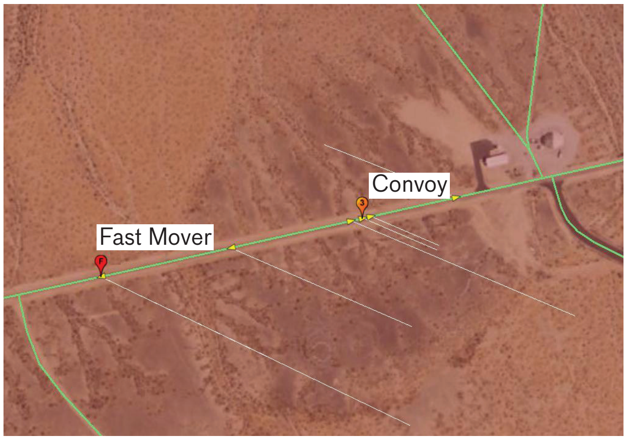


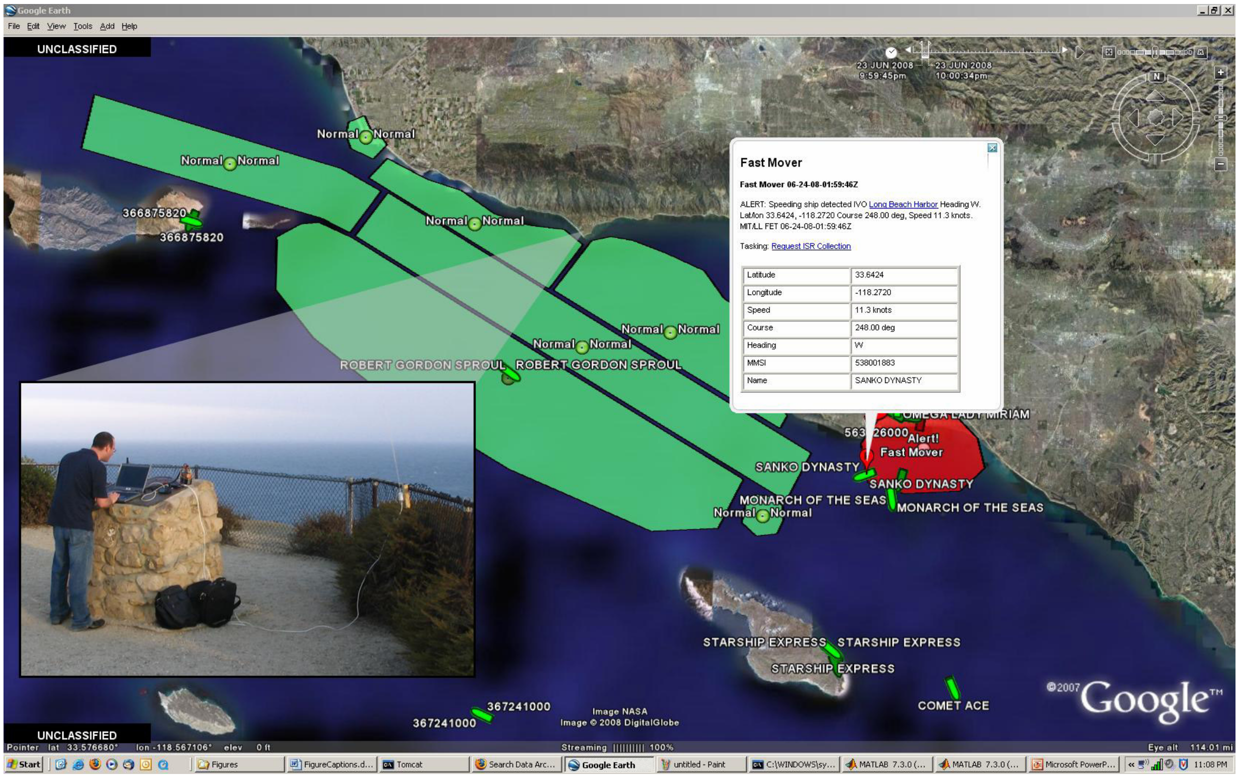
FIGURE 7. Projected MTI detections provide enhanced situational awareness. The raw position of the MTI detections are at the ends of the white lines. Without the FET, it would be difficult to interpret where the vehicles really were and in what direction they were heading. The FET projects these detections onto the nearest road and displays a triangle representing the estimated location and direction of travel of each vehicle. The color of the triangles is based on the signal strength of the detections.

Figure 7 shows the MTI detections projected onto a road network. The triangles show direction of travel. All the detections are yellow, indicating high-SNR returns—that is, likely vehicles. The ends of the white lines correspond to the original dot location, and the white line indicates the projection along the cross-range error to the nearest road. Notice also the orange paddle with the number 3 in it, labeled "Convoy" to indicate the detection of a three-vehicle convoy by the FET. (A convoy is defined as two or more vehicles traveling in the same direction, with approximately the same speed and relatively small inter-vehicle spacing.)

The other types of exploitation the FET performs on MTI are determination of fast movers and slow movers (vehicles moving at speeds higher or lower than the average); higher or lower level of activity, relative to a predefined norm; and the presence of an abnormally large convoy or of an abnormally large number of convoys. All of these alerts cause the watchbox region to turn red. In addition, the fast- and slow-mover alerts and convoy detections display an alert paddle showing their location. While the FET allows the user to display raw MTI, the display is often easier to understand when only the projected dots and exploitation results are shown.

Fusion

In the ISR community, fusion has two broad definitions. The first is the ability to have a large variety of information available for display in distinct layers within the same application. Such layers may include the underlying earth globe and maps, road networks, air traffic, ground vehicle activity, tracks, and exploitation results. Users



Fast Mover

Fast Mover 06-24-08-01:59:46Z

ALERT: Speeding ship detected IVO Long Beach Harbor Heading W. Lat/lon 33.6424, -118.2720 Course 248.00 deg, Speed 11.3 knots. MIT/LL FET 06-24-08-01:59:46Z

Tasking: Request ISR Collection

Latitude	33.6424
Longitude	-118.2720
Speed	11.3 knots
Course	248.00 deg
Heading	W
MMSI	538001883
Name	SANKO DYNASTY

FIGURE 8. During a field data collection from Point Dume State Beach in Malibu, Calif., in June 2008, the FET recorded and ingested data from the maritime Automatic Identification System (AIS). One of the ships (the *Sanko Dynasty*) was detected as a fast mover, with a speed faster than the normal speed of maritime traffic in Long Beach Harbor. This alert caused a warning state to be activated, indicated by the Long Beach Harbor watchbox (shaded red), alerting the user to abnormal activity in that area.

can turn on (or off) the layers that they're interested in. This concept is referred to as a user-defined operational picture (UDOP). Google Earth, with its ability to easily present a large variety of layers of information under user control, has emerged as the preferred backbone of a UDOP system.

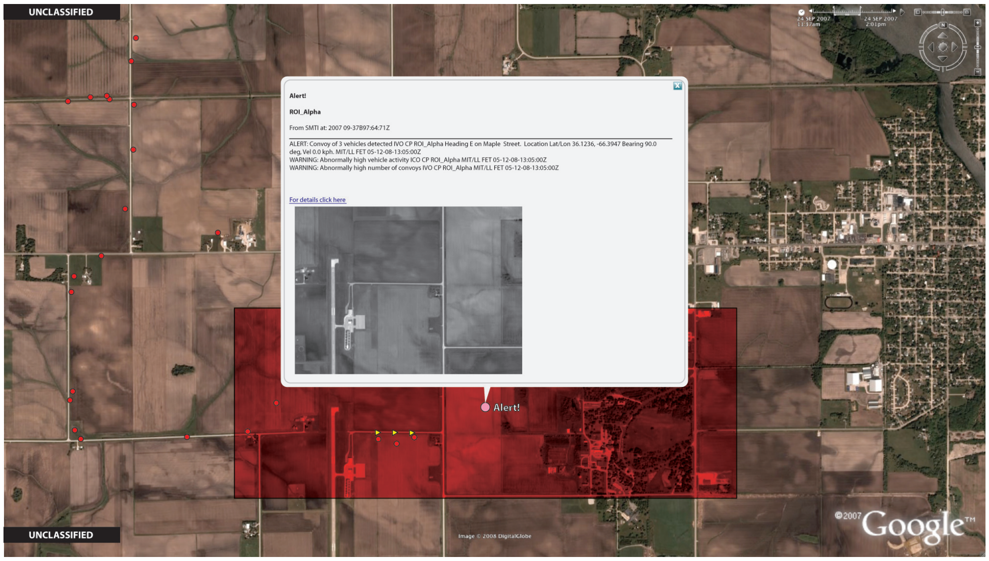
The second definition of fusion refers to software algorithms that operate on various types of ISR data. There are several ways in which different ISR data types are fused by the FET to get a more coherent understanding of the surveillance area. The sections that follow describe two additional ISR data types—the maritime Automatic Identification System and the Constant Hawk persistent surveillance system—and how the FET processes their data.

Automatic Identification System

An emerging standard for the display of maritime vessel information is the Automatic Identification System, or AIS. AIS is a set of radio-frequency messages that indicate the identification of a ship, its intended port, its estimated time of arrival, and its present position, heading, and velocity.

The FET reads AIS messages and displays them in Google Earth; it also performs exploitation analysis on the data. Maritime watchboxes are defined, and AIS activity in the area is analyzed. The algorithms evaluate higher than normal (and lower than normal) activity and perform the same type of fast-mover and slow-mover detection as for MTI. Another function generates an alert if several ships appear to rendezvous. Figure 8 shows AIS data collected off Point Dume, Calif., and exploited by the FET, with a fast-mover alert highlighted.

An additional algorithm takes into account both AIS and MTI data. This multi-intelligence fusion algorithm attempts to detect false AIS position reports. Such a situation could arise if a nefarious ship doesn't want its true position known and so starts sending out false position information. The FET uses radar MTI information to confirm (or reject) the presence of a ship in the vicinity of



Alert!

ROI_Alpha

From SMTI at: 2007 09-37B97:64:71Z

ALERT: Convoy of 3 vehicles detected IVO CP ROI_Alpha Heading E on Maple Street. Location Lat/Lon 36.1236, -66.3947 Bearing 90.0 deg, Vel 0.0 kph. MIT/LL FET 05-12-08-13:05:00Z
WARNING: Abnormally high vehicle activity ICO CP ROI_Alpha MIT/LL FET 05-12-08-13:05:00Z
WARNING: Abnormally high number of convoys IVO CP ROI_Alpha MIT/LL FET 05-12-08-13:05:00Z

For details click here

FIGURE 9. The results of the Constant Hawk video MTI exploitation are enhanced with short snippets of video (called video chips), displayed in a pop-up window in this notional example.

an AIS report. The FET does so by looking for high-SNR MTI detections in the vicinity of an AIS position report. If the AIS position can be corroborated with the MTI data, then the FET indicates it by using the same type of paddle icon that it displays for convoys, fast movers, and other activity of concern.

Constant Hawk Integration

Constant Hawk is a wide-area, large-format sensor used for collecting large swaths of imagery (roughly 12 by 12 km), updated twice per second. This imagery product is used for viewing persistent surveillance video in various high-conflict areas of the world. The imagery is of high enough resolution that individual vehicles can be detected and tracked. The FET integrates Constant Hawk data in several ways.

The Constant Hawk system performs image processing algorithms on the data it collects and is able to identify moving vehicles in its video. The FET converts these Constant Hawk detections into the STANAG 4607 format so that they can then be treated in the same fashion as any other MTI. This standardization allows the FET to perform the same automated exploitation of the video MTI that it does on the radar MTI. When an alert is detected, FET generates, on the fly, a short video snippet (known as a video chip) in the direct vicinity of the alert, with a time span of plus or minus 10 seconds from when the alert was generated. These videos have the same Macromedia Flash format (.swf) as YouTube videos. Such compatibility is important because it allows the FET to generate pop-up alert windows in Google Earth that contain the Constant Hawk video chip. Because this video chip is a Flash movie, the user can pan, zoom, rewind, and replay, just as with any other Flash movie (Figure 9).

The other way that the FET has integrated Constant Hawk data is by supporting the display of the high-resolution Constant Hawk image still frames in Google Earth (Figure 10). Normally, high-resolution imagery is built directly into the Google Earth ground plane—a time-consuming process that requires the Google Earth data brick to be updated on a centralized server. Alternatively, the data can be downsampled and a lower-resolution version displayed. The FET, though, has incorporated multiresolution tile processing for display of these high-resolution images. Also known as image pyramids, or mip-maps, such processing displays low-resolution versions if the viewer's perspective is far away from the image. As the viewer gets closer and closer, higher-resolution tiles are paged in.

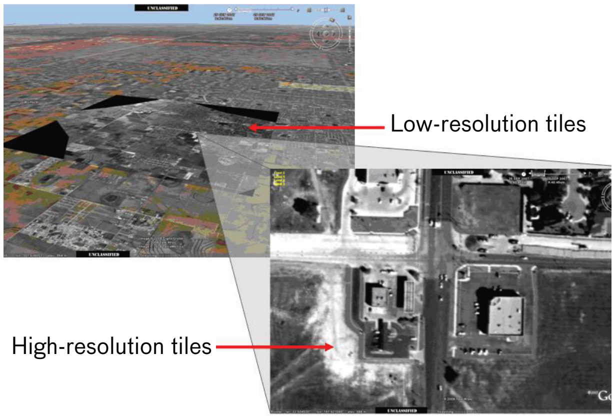


FIGURE 10. Constant Hawk imagery displayed in Google Earth: as the viewer zooms in, higher-resolution images are paged in to replace lower-resolution ones.

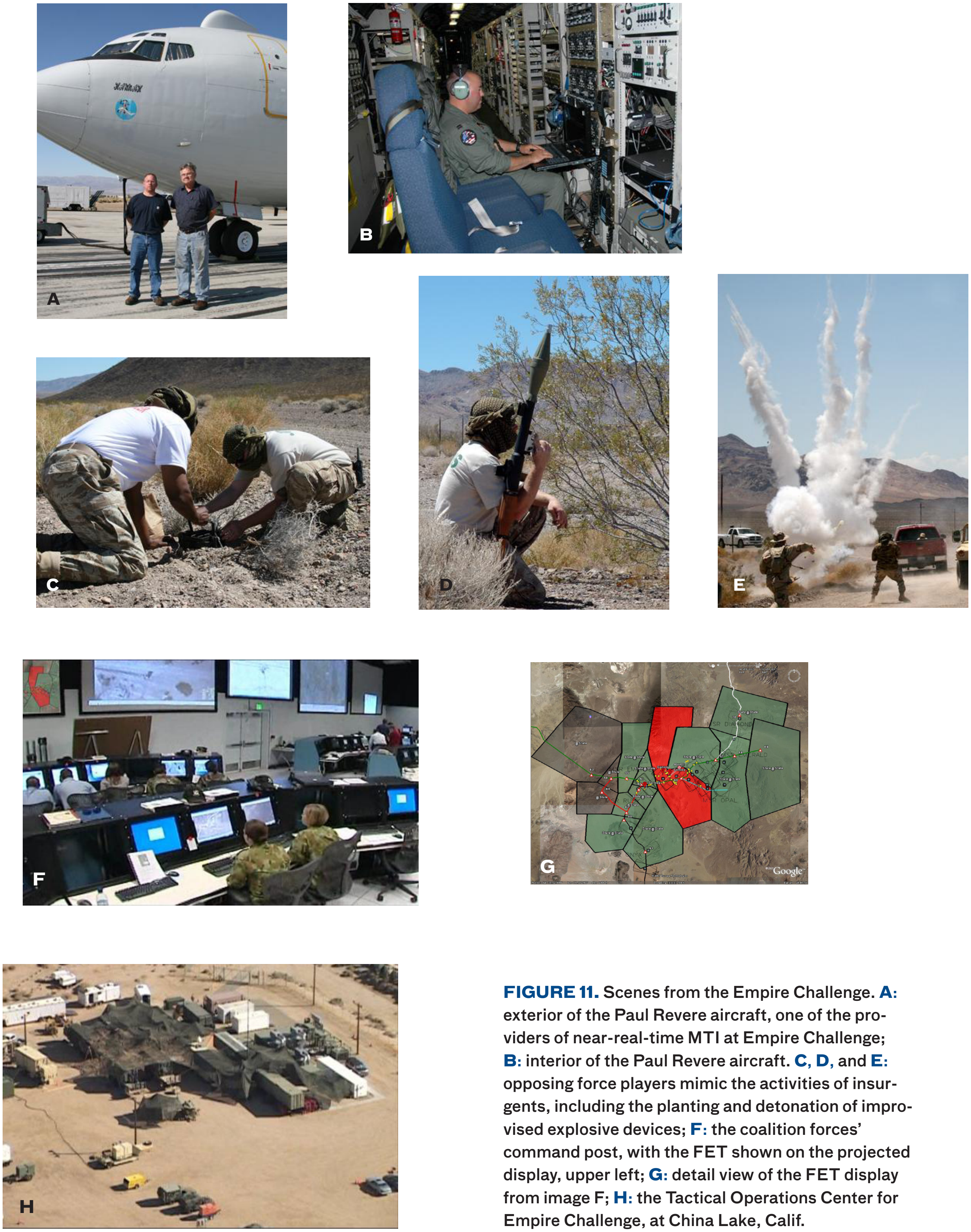


FIGURE 11. Scenes from the Empire Challenge. **A:** exterior of the Paul Revere aircraft, one of the providers of near-real-time MTI at Empire Challenge; **B**: interior of the Paul Revere aircraft. **C, D,** and **E:** opposing force players mimic the activities of insurgents, including the planting and detonation of improvised explosive devices; **F:** the coalition forces' command post, with the FET shown on the projected display, upper left; **G:** detail view of the FET display from image F; **H:** the Tactical Operations Center for Empire Challenge, at China Lake, Calif.

Empire Challenge

The NGA, in partnership with the Joint Forces Command (JFCOM), has been running a series of field experiments called Empire Challenge. For the past few years, Empire Challenge has been taking place at the China Lake Naval Weapons Station in Ridgecrest, Calif., in the northeast corner of the Mojave Desert. With a focus on ISR sensors, systems, and integration, and participation by coalition partners (Britain, Canada, and Australia), it has become the premier annual intelligence-related field exercise.

Lincoln Laboratory has been participating in Empire Challenge for several years, flying the Paul Revere aircraft with LiMIT radar and instrumenting ground vehicles with GPS units in order to collect ground truth. Starting with Empire Challenge 2007 (EC07), the FET began participating in the exercise (Figure 11).

At EC07, the FET had a limited presence that marked the debut of the FET outside of Lincoln Laboratory; it was essentially a proof of concept. This initial release of the FET performed well. Moreover, the design choice of going with a network-centric architecture (requiring only an Internet browser and the Google Earth client to run the FET) paid off handsomely: several organizations at EC07 discovered the FET on the network and used it as an additional situation-awareness display. Two particular examples of users of this initial FET release at EC07 are the NGA GeoScout Concept Car project and the Canadian All-Source Intelligence Center.

Participation in EC07 made it clear that for FET to be a serious contender in a tactical system, it would have to be modified with an emphasis on scalability and real-time operation. In particular, it would need to be able to keep up with a live stream of MTI data from the JSTARS aircraft. The other lesson was that full-motion video from a ScanEagle UAV was the dominant form of intelligence, and an effort would need to be made to better integrate ScanEagle with the FET.

At Empire Challenge 2008, which concluded on August 1, FET was a huge success. In particular, our recent work on scalability yielded positive results: on the final day of the exercise, the FET consumed and exploited—in real time—591,641 MTI detections from JSTARS.

The FET was run by numerous users in several rooms at China Lake. In addition, it was run locally on Hanscom Air Force Base at the Electronic Systems Center Command and Control Enterprise Integration Facility, with a direct connection to the China Lake network, where the FET server was located. FET data were also sent out over SIPRNet to a remote site as well as to Lincoln Laboratory, allowing us to provide support from a variety of locations.

The FET was used to directly support the commander of U.S. forces (Task Force Lancer) as part of the Joint Battlespace Awareness ISR Integration Capability effort. FET provided situation awareness that was available through no other means. It also routinely protected U.S. ground forces and provided tips and cues to the ScanEagle, including several examples of direct machine-to-machine tasking. The Task Force Lancer commander stated that the FET added "tremendous situation awareness" to his tactical view of the battlefield.

With the positive feedback we have received from our participation in Empire Challenge 2008, our main goal for the upcoming year is to identify an operational site or tactical unit willing to serve as an alpha site to explore the use of the FET within an operational ISR community. In addition to continued refinement of the exploitation algorithms and work on performance characterization, we are also hoping to add the ability to display and exploit signals intelligence (SIGINT). At EC08, SIGINT reports often gave key clues regarding (simulated) insurgent activity. SIGINT would be a complementary data type to both MTI and AIS, and would allow a variety of multi-intelligence exploitation algorithms to be investigated.

Acknowledgments

Lt. Col. Roger Claypoole was the original force driving us to develop innovative solutions in automated MTI exploitation. Roger's right-hand man through most of the FET development was Bob Lanahan of Mantech SRS. At EC08, Bob was the lead of the Multi-Domain Awareness thread, incorporating both maritime scenarios off the coast of California and ground scenarios taking place at China Lake. Lincoln Laboratory management provided valuable support and encouragement; special thanks go to FET program manager Stephen Pohlig and to Doug Marquis, who played a key role in helping get the AIS data feed up and running at the Laboratory.

The FET wouldn't be where it is today without the crack team of developers working on it, none of whom accept the phrase "no, that can't be done" as an acceptable answer from anyone: Tim Schreiner, who played a key role in the initial FET release; Bill Ledder and John Nwagbaraocha, who took on many development tasks and who put the system through rigorous testing; Peter Mastromarino and Ryan Holt, the principal developers of the exploitation algorithms; and Jim Garlick and Tod Shannon, the surgeons who work behind the scenes and keep the heart of the FET running (and who also moonlight as software assassins, able to eliminate any bug). Without their contributions, the FET would still just be "the little science project in the corner." ■

ABOUT THE AUTHOR

Paul Metzger is a staff member in Lincoln Laboratory's Intelligence and Decision Technologies Group. He is the originator of the FET concept and has been the senior architect of the system since its inception, as well as a technical contributor on various components. Before joining Lincoln Laboratory, Metzger was a founder and chief engineer at Reality by Design, which provided immersive 3D training environments to the Department of Defense. He has a bachelor's degree in electrical engineering from Northeastern University.